What is claimed is:

1. A method for updating coefficients in a decision feedback equalizer with an ISI canceller for canceling ISI from a plurality of first signals received from a channel, the method comprising:

decoding a first symbol comprising a set of the first signals to generate a decoded symbol, wherein the first symbol has (k+1) chips, and k is natural number;

obtaining a vector of error values computed as the difference between the decoded symbol, and the first symbol;

generating a temp matrix according to the decoded symbol and the vector of the error values;

averaging the values of the elements in every diagonal line of the temp matrix to generate a Toeplitz Matrix; and

updating the coefficients by the Toeplitz Matrix.

2. The method as claimed in claim 1 further comprises:

updating coefficients according to a least mean square algorithm:

$H(m+1)=H(m)+\mu T\{\mathrm{conj}(E(m))\bullet C(m+1)\}$;

H(m)is coefficients at a symbol time m;

H(m+1) is coefficients at a symbol time (m+1);

ì is a predetermined gain;

T is the Toeplitz Matrix;

E(m)is the vector of error values; and

C(m+1) is the decoded symbol at the symbol time (m+1).

3. The method as claimed in claim 1, wherein, in the Toeplitz Matrix $\begin{bmatrix} h_{(k+1)} & h_{(k+2)} & \cdots & \cdots & h_{(2k+1)} \\ h_k & h_{(k+1)} & \cdots & \cdots & h_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ h_2 & h_3 & \cdots & \cdots & h_{(k+2)} \\ h_1 & h_2 & \cdots & \cdots & h_{(k+1)} \end{bmatrix}$, for any $(2k+1) \geq i > (k+1)$, the $h_{(i)} \ldots h_{(2k+1)}$ are equal to 0.

4. A method for updating coefficients in a decision feedback equalizer with an ISI canceller for canceling ISI from a plurality of first signals received from a channel, the method comprising:

decoding a first symbol comprising a set of the first signals to generate a decoded symbol, wherein the first symbol has (k+1) chips, and k is natural number;

obtaining a vector of error values computed as the difference between the decoded symbol, and the first symbol;

generating a temp Matrix T(m) according to the decoded symbol and the vector of the error values, wherein T(m)=

$$\begin{bmatrix} E^*(n-k)\cdot C(n-k) & E^*(n-k)\cdot C(n-(k-1)) & \cdots & \cdots & E^*(n-k)\cdot C(n) \\ E^*(n-(k-1))\cdot C(n-k) & E^*(n-(k-1))\cdot C(n-(k-1)) & \cdots & \cdots & E^*(n-(k-1))\cdot C(n) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ E^*(n-1)\cdot C(n-k)) & E^*(n-1)\cdot C(n-(k-1)) & \cdots & \cdots & E^*(n-1)\cdot C(n) \\ E^*(n)\cdot C(n-k) & E^*(n)\cdot C(n-(k-1)) & \cdots & \cdots & E^*(n)\cdot C(n) \end{bmatrix},$$

where m is the symbol time of the first symbol, the chip times of the first symbol are from (n-k)

to n, n and m are natural numbers and n=(k+1)m; E(n) is a vector of error values at the chip time n; and C(n) is the chip of the decoded symbol at the chip time n;

averaging the values of the elements in every diagonal line of the temp matrix to generate a Toeplitz

Matrix $\begin{bmatrix} h_{(k+1)} & h_{(k+2)} & \cdots & \cdots & h_{(2k+1)} \\ h_k & h_{(k+1)} & \cdots & \cdots & h_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ h_2 & h_3 & \cdots & \cdots & h_{(k+2)} \\ h_1 & h_2 & \cdots & \cdots & h_{(k+1)} \end{bmatrix}$, wherein H(m)=

$$\begin{bmatrix} (\sum_{i=0}^{k} E^*(n-i)\cdot C(n-i))/(k+1) & (\sum_{i=0}^{k-1} E^*(n-(i+1))\cdot C(n-i))/k & \cdots\cdots & E^*(n-k)\cdot C(n) \\ (\sum_{i=0}^{k-1} E^*(n-i)\cdot C(n-(i+1))/k & (\sum_{i=0}^{k} E^*(n-i)\cdot C(n-i))/(k+1) & \cdots\cdots & (\sum_{i=0}^{k-(k-1)} E^*(n-(i+k-1))\cdot C(n-i))/2 \\ \vdots & \vdots & \vdots\ \vdots & \vdots \\ (\sum_{i=0}^{k-(k-1)} E^*(n-i)\cdot C(n-(i+k-1)))/2 & (\sum_{i=0}^{k-(k-2)} E^*(n-i)\cdot C(n-(i+k-2)))/3 & \cdots\cdots & (\sum_{i=0}^{k-1} E^*(n-(i+1))\cdot C(n-i))/k \\ E^*(n)\cdot C(n-k) & (\sum_{i=0}^{k-(k-1)} E^*(n-i)\cdot C(n-(i+k-1)))/2 & \cdots\cdots & (\sum_{i=0}^{k} E^*(n-i)\cdot C(n-i))/(k+1) \end{bmatrix}$$

where H(m) is the Toeplitz Matrix at the symbol time m.

5. The method as claimed in claim 4 further comprises:

updating coefficients according to a least mean square algorithm:

$H(m+1)=H(m)+\mu T\{conj(E(m))\bullet C(m+1)\}$;

H(m)is coefficients at a symbol time m;

H(m+1) is coefficients at a symbol time (m+1);

ì is a predetermined gain;

T is the Toeplitz Matrix;

E(m) is the vector of error values; and C(m+1) is the decoded symbol at the symbol time (m+1).

6. The method as claimed in claim 4, wherein, in the Toeplitz Matrix

$$\begin{bmatrix} h_{(k+1)} & h_{(k+2)} & \cdots & \cdots & h_{(2k+1)} \\ h_k & h_{(k+1)} & \cdots & \cdots & h_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ h_2 & h_3 & \cdots & \cdots & h_{(k+2)} \\ h_1 & h_2 & \cdots & \cdots & h_{(k+1)} \end{bmatrix}$$

, for any $(2K+1) \geq i > (k+1)$, the $h_{(i)} \ldots h_{(2k+1)}$ are equal to 0.

7. A decision feedback equalizer, comprising:

an ICI canceller for canceling ICI from a signal received from a channel and outputting a first signal without ICI; and

an ISI canceller, comprising:

a symbol decoder for decoding a first symbol comprising a set of the first signals to generate a decoded symbol; and

a symbol-base feedback filter with a plurality coefficients for transforming the decoded symbol by a Toeplitz Matrix **H**(m) to cancel ISI from the present decoded symbol, and generating an output signal;

wherein the first symbol has (k+1) chips, the Toeplitz Matrix is a (k+1)*(k+1) matrix, m is the symbol time of the first symbol, the chip times of the first symbol are from (n-k) to n, n, k and m are natural numbers and n=(k+1)m;

$$H(m)=\begin{bmatrix}
(\sum_{i=0}^{k}E^{*}(n-i)\cdot C(n-i))/(k+1) & (\sum_{i=0}^{k-1}E^{*}(n-(i+1))\cdot C(n-i))/k & \cdots\cdots & E^{*}(n-k)\cdot C(n) \\
(\sum_{i=0}^{k-1}E^{*}(n-i)\cdot C(n-(i+1)))/k & (\sum_{i=0}^{k}E^{*}(n-i)\cdot C(n-i))/(k+1) & \cdots\cdots & (\sum_{i=0}^{k-(k-1)}E^{*}(n-(i+k-1))\cdot C(n-i))/2 \\
\vdots & \vdots & \vdots\ \vdots & \vdots \\
(\sum_{i=0}^{k-(k-1)}E^{*}(n-i)\cdot C(n-(i+k-1)))/2 & (\sum_{i=0}^{k-(k-2)}E^{*}(n-i)\cdot C(n-(i+k-2)))/3 & \cdots\cdots & (\sum_{i=0}^{k-1}E^{*}(n-(i+1))\cdot C(n-i))/k \\
E^{*}(n)\cdot C(n-k) & (\sum_{i=0}^{k-(k-1)}E^{*}(n-i)\cdot C(n-(i+k-1)))/2 & \cdots\cdots & (\sum_{i=0}^{k}E^{*}(n-i)\cdot C(n-i))/(k+1)
\end{bmatrix}$$

where E(n) is a vector of error values computed as the difference between the chip of the decoded symbol at the chip time n, and the chip input to the symbol decoder at the chip time n, and C(n) is the chip of the decoded symbol at the chip time n.

8. The decision feedback equalizer as claimed in claim 7, wherein the coefficients are updated according to a least mean square algorithm:

$H(m+1)=H(m)+\mu T\{conj(E(m))\bullet C(m+1)\}$;

H(m) is coefficients at a symbol time m;

H(m+1) is coefficients at a symbol time (m+1);

ì is a predetermined gain;

T{} is the Toeplitz Matrix;

E(m) is the vector of error values; and

C(m+1) is the decoded symbol at the symbol time (m+1).

9. The decision feedback equalizer as claimed in claim 7, wherein the Toeplitz Matrix

$$\begin{bmatrix} h_{(k+1)} & h_{(k+2)} & \cdots & \cdots & h_{(2k+1)} \\ h_k & h_{(k+1)} & \cdots & \cdots & h_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ h_2 & h_3 & \cdots & \cdots & h_{(k+2)} \\ h_1 & h_2 & \cdots & \cdots & h_{(k+1)} \end{bmatrix}$$ at the symbol time m is

$$\begin{bmatrix} E^*(n-k)\cdot C(n-k) & E^*(n-k)\cdot C(n-(k-1)) & \cdots & \cdots & E^*(n-k)\cdot C(n) \\ E^*(n-(k-1))\cdot C(n-k) & E^*(n-(k-1))\cdot C(n-(k-1)) & \cdots & \cdots & E^*(n-(k-1))\cdot C(n) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ E^*(n-1)\cdot C(n-k)) & E^*(n-1)\cdot C(n-(k-1)) & \cdots & \cdots & E^*(n-1)\cdot C(n) \\ E^*(n)\cdot C(n-k) & E^*(n)\cdot C(n-(k-1)) & \cdots & \cdots & E^*(n)\cdot C(n) \end{bmatrix}$$, and when the channel is steady, the values of the elements in the diagonal lines of the Toeplitz Matrix are almost the same, $h_{11}=h_{22}=\ldots=h_{(k+1)(k+1)}$, $h_{21}=h_{32}=\ldots=h_{(k+1)k}$, ... , $h_{k1}=h_{(k+1)2}$, $h_{12}=h_{23}=\ldots=h_{k(k+1)}$, $h_{13}=h_{24}=\ldots=h_{kk}$, ..., $h_{1k}=h_{2(k+1)}$.

10. The decision feedback equalizer as claimed in claim 7, wherein, in the Toeplitz Matrix $$\begin{bmatrix} h_{(k+1)} & h_{(k+2)} & \cdots & \cdots & h_{(2k+1)} \\ h_k & h_{(k+1)} & \cdots & \cdots & h_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ h_2 & h_3 & \cdots & \cdots & h_{(k+2)} \\ h_1 & h_2 & \cdots & \cdots & h_{(k+1)} \end{bmatrix}$$, for any $(2k+1) \geq i > (k+1)$, the $h_{(i)} \ldots h_{(2k+1)}$ are equal to 0.